

Searching PAJ

1/2 ページ

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-044460

(43)Date of publication of application : 14.02.1997

(51)Int.Cl.

G06F 15/16

(21)Application number : 07-191742

(71)Applicant : NEC CORP

(22)Date of filing : 27.07.1995

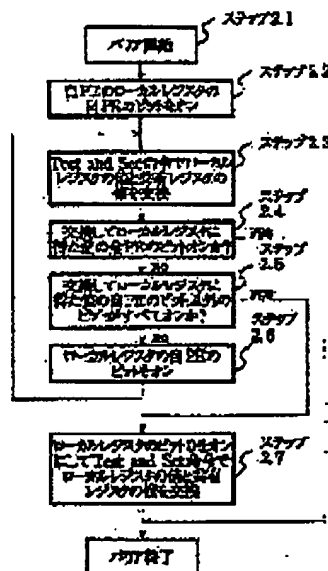
(72)Inventor : YAMADA SHOICHIRO

(54) BARRIER SYNCHRONIZATION PROCESSING METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To provide barrier synchronization at a high speed even when the number of processors is increased.

SOLUTION: Respective PEs turn on the bit of the present PE of the local register of the present PE (a step 2.2). Then, the value of the local register and the value of a shared register are exchanged by a TS instruction (the step 2.3). Then, whether or not the bits of all the PEs of the value obtained in the local register by exchange are on is checked (the step 2.4). When all are on, it is considered as a barrier synchronization end. When all bits are not on, whether or not the bits except the bit of the present PE of the local register are all on is checked (the step 2.5). When the step 2.5 is no, the bit of the present PE of the local register is turned on (the step 2.6) and the step 2.3 is returned. When the step 2.5 is yes, the bit '0' of the local register is turned on and the value of the local register and the value of the shared register are exchanged by the TS instruction (the step 2.7).



LEGAL STATUS

[Date of request for examination] 27.07.1995

[Date of sending the examiner's decision of rejection] 13.10.1998

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-44460

(43) 公開日 平成9年(1997)2月14日

(51) Int. Cl.⁵

G 0 6 F 15/16

識別記号

3 5 0

庁内整理番号

F I

G 0 6 F 15/16

技術表示箇所

3 5 0 Z

審査請求 有 請求項の数 1 O L (全 6 頁)

(21) 出願番号 特願平7-191742

(22) 出願日 平成7年(1995)7月27日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 山田 正一郎

東京都港区芝五丁目7番1号 日本電気株式会社内

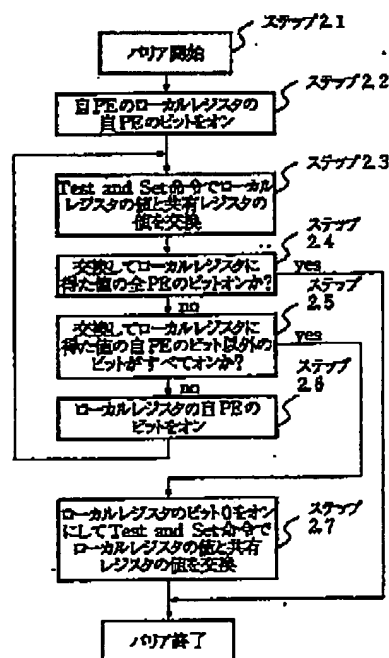
(74) 代理人 弁理士 京本 直樹 (外2名)

(54) 【発明の名称】 バリア同期処理方法

(57) 【要約】

【課題】 プロセッサ台数が増えても高速にバリア同期を実現する。

【解決手段】 各々のP Eは、自P Eのローカルレジスタの自P Eのビットをオンにする(ステップ2.2)。次に、TS命令でローカルレジスタの値と共有レジスタの値を交換する(ステップ2.3)。次に、交換によりローカルレジスタに得た値の全P Eのビットがオンかチェックする(ステップ2.4)。全てがオンであればバリア同期終了とみなす。全てのビットがオンでなければ、ローカルレジスタの自P Eのビット以外のビットがすべてオンかチェックする(ステップ2.5)。ステップ2.5がnoであれば、ローカルレジスタの自P Eのビットをオンにし(ステップ2.6)、ステップ2.3に戻る。ステップ2.5でyesであれば、ローカルレジスタのビット0をオンにしてTS命令でローカルレジスタの値と共有レジスタの値を交換する(ステップ2.7)。



(2)

特開平9-44460

1

【特許請求の範囲】

【請求項1】 n 個 (n は2以上の整数)のプロセッサと共有レジスタから構成され前記プロセッサには各々ローカルレジスタを有するマルチプロセッサシステムにおけるバリア同期処理方法において、ビット0はロックビット、ビット1からビット n までの各ビットは各々前記 n 個のプロセッサの1つに対応するビット構成を有する前記共有レジスタと前記ローカルレジスタを予め用意し、各々の前記プロセッサが、自プロセッサの前記ローカルレジスタの自プロセッサ対応のビットをオンにし (ステップ2.2)、次に、TS (Test & Set) 命令で前記ローカルレジスタの情報と前記共有レジスタの情報を交換し (ステップ2.3)、次に、前記交換により前記ローカルレジスタに得た前記情報における全プロセッサのビットがオンかチェックし (ステップ2.4)、前記全プロセッサのビットが全てオンであればバリア同期終了とみなし、前記全プロセッサのビットにおける1つのビットでもオンでなければ前記ローカルレジスタにおける自プロセッサのビット以外のビットがすべてオンかチェックし (ステップ2.5)、前記ステップ2.5におけるチェック結果すべてオンでなければ前記ローカルレジスタの前記自プロセッサのビットをオンにし (ステップ2.6) た後、前記ステップ2.3に戻り、前記ステップ2.5におけるチェック結果すべてオンであれば前記ローカルレジスタの前記ビット0をオンにして前記TS命令で前記ローカルレジスタの情報と前記共有レジスタの情報を交換する (ステップ2.7) ことを特徴とするバリア同期処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、計算機システム特にマルチプロセッサシステムにおけるバリア同期処理方法に関する。

【0002】

【従来の技術】 従来のマルチプロセッサシステムにおけるバリア同期処理方法では、Test And Set 命令を利用していた。Test And Set命令 (以降、TS 命令と記す) とは、自プロセッサのローカルレジスタの値と共有レジスタの値を交換する命令で、共有レジスタの最上位ビットがオンの場合には、共有レジスタの値のローカルレジスタへのコピーのみを行う。この命令を使用することによって、共有レジスタのロックおよび変更を行う。

【0003】 図1はマルチプロセッサシステムのシステム構成図、図4は共有レジスタおよびローカルレジスタの各ビットの意味付けを説明した図、図5は従来のバリア同期処理の処理フロー図である。

【0004】 まず、それぞれのプロセッサ (以降PEと記す) がロックを示すビット0をオンにした状態でスタートする (ステップ4.1)。PE1がTS命令を発行し、共有レジスタとの値の交換を行う (ステップ4.

2

2)。コピーされた値のロックビットのビット0がオフなので、ロックがPE1に与えられる (ステップ4.3)。PE1が自PEのビットであるビット1をオンにして共有レジスタへ書き込みを行う (ステップ4.4)。ここでの書き込みには既にロックを得ているので、通常の命令を使用して強制的に書き込む。次にPE2がTS命令を発行して共有レジスタとの値の交換を行う (ステップ4.5)。コピーされた値のロックビットのビット0がオフなので、ロックがPE2に与えられる (ステップ4.6)。PE2が自PEのビットであるビット2をオンにして共有レジスタへ書き込みを行う (ステップ4.7)。これで、共有レジスタ内のビットが全てオンになるので、バリア動機が成立する (ステップ4.8)。

【0005】

【発明が解決しようとする課題】 上述した従来のバリア同期処理方法では、ロックを使用してバリア同期処理を行うため、ロックが解除されるまでの他のプロセッサが全て待たされてバリア同期の処理に時間がかかるという欠点があり、特にプロセッサ台数が多い場合に処理時間の増加が顕著であり、マルチプロセッサの性能が低下するという問題がある。

【0006】

【課題を解決するための手段】 本発明は、 n 個 (n は2以上の整数) のプロセッサと共有レジスタから構成され前記プロセッサには各々ローカルレジスタを有するマルチプロセッサシステムにおけるバリア同期処理方法において、ビット0はロックビット、ビット1からビット n までの各ビットは各々前記 n 個のプロセッサの1つに対応するビット構成を有する前記共有レジスタと前記ローカルレジスタを予め用意し、各々の前記プロセッサが、自プロセッサの前記ローカルレジスタの自プロセッサ対応のビットをオンにし (ステップ2.2)、次に、TS (Test & Set) 命令で前記ローカルレジスタの情報と前記共有レジスタの情報を交換し (ステップ2.3)、次に、前記交換により前記ローカルレジスタに得た前記情報における全プロセッサのビットがオンかチェックし (ステップ2.4)、前記全プロセッサのビットが全てオンであればバリア同期終了とみなし、前記全プロセッサのビットにおける1つのビットでもオンでなければ前記ローカルレジスタにおける自プロセッサのビット以外のビットがすべてオンかチェックし (ステップ2.5)、前記ステップ2.5におけるチェック結果すべてオンでなければ前記ローカルレジスタの前記自プロセッサのビットをオンにし (ステップ2.6) た後、前記ステップ2.3に戻り、前記ステップ2.5におけるチェック結果すべてオンであれば前記ローカルレジスタの前記ビット0をオンにして前記TS命令で前記ローカルレジスタの情報と前記共有レジスタの情報を交換する (ステップ2.7) ことを特徴とする。

(3)

特開平9-44460

3

【 0007】

【 発明の実施の形態】次に、本発明について図面を参照して説明する。

【 0008】図1 は本発明のバリア同期処理方法が実施されるシステム構成図であり、図2 は本発明の一実施例を示すバリア同期処理フロー図、図3 は本発明のバリア同期処理による動作説明図である。

【 0009】以下の例では、2 プロセッサの場合について説明するが、3 プロセッサ以上の場合についても同様である。

【 0010】本発明においても、ローカルレジスタと共有レジスタの値を交換する命令としてTS (Test And Set) 命令を使用する。即ち、共有レジスタの最上位ビットがオフならば、TS 命令は単純な交換命令として動作する。

【 0011】図2 を参照すると、各々のPE は、自PE のローカルレジスタの自PE のビットをオンにする (ステップ2. 2) 。次に、TS 命令でローカルレジスタの値と共有レジスタの値を交換する (ステップ2. 3) 。次に、交換によりローカルレジスタに得た値の全PE のビットがオンかチェックする (ステップ2. 4) 。全てがオンであればバリア同期終了とみなしバリア終了となる (ステップ2. 8) 。全てのビットがオンでなければ、ローカルレジスタの自PE のビット以外のビットがすべてオンかチェックする (ステップ2. 5) 。ステップ2. 5 がno であれば、ローカルレジスタの自PE のビットをオンにし (ステップ2. 6) 、ステップ2. 3 に戻る。ステップ2. 5 でyes であれば、ローカルレジスタのビット0 をオンにしてTS 命令でローカルレジスタの値と共有レジスタの値を交換して (ステップ2. 7) 、ステップ2. 8 に進む。

【 0012】次に、図3 を用いて具体例を説明する。まず、それぞれのPE (プロセッサ) が自PE 対応のビットをオン (論理1) にした状態からスタートする (ステップ3. 1) 。PE1 がTS 命令を発行して (ステップ3. 2) 、共有レジスタの値とローカルレジスタの値を交換する (ステップ3. 3) 。ここでは、ロックビットをオンにはしていないので、共有レジスタのロックはされず、共有レジスタのアクセスはどのPE からでも自由

4

にできる。次に、PE2 がTS 命令を発行してローカルレジスタと共有レジスタの値を交換する (ステップ3. 4) 。PE2 は、ローカルレジスタにコピーされた値の自PE のビットをオンにする (ステップ3. 6) 。ここで、全PE のビットがオンになったので、ロックビットもオンにする (ステップ3. 7) 。これによって、全ビットがオンになった値が共有レジスタから交換されてしまうのを防止する。PE2 がTS 命令を発行して共有レジスタと値を交換する (ステップ3. 8) 。これで共有レジスタのビットが全てオンになったのでバリア同期が成立する。ロックビットがオンなので、他のPE からのTS 命令の発行にも書き換えられることなく、他のPE もこの値を参照してすぐにバリアが成立する。

【 0013】この方法によるTS 命令の発行回数は、PE 台数に対して穏かに上昇して、従来技術に比べるとPE 台数が増えれば増えるほど効果が顕著となる。

【 0014】

【 発明の効果】以上説明したように、本発明のバリア同期処理方法は、ロックを使用しないで共有レジスタとローカルレジスタの値を交換することを主にバリア同期を実現するため、各プロセッサが常に共有レジスタを定義参照でき、プロセッサ台数が増えても高速にバリア同期が実現できるという効果がある。

【 図面の簡単な説明】

【 図1】バリア同期処理が適用されるシステム構成例を示す図である。

【 図2】本発明のバリア同期処理の一実施例を示すフローチャートである。

【 図3】本発明の動作を説明するためのフローチャートである。

【 図4】共有レジスタの構成を示す図である。

【 図5】従来例のバリア同期処理方法の一例を示すフローチャートである。

【 符号の説明】

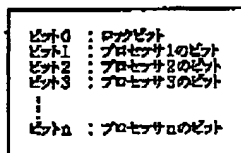
1. 1, 1. 4 PE (プロセッサ)

1. 3 共有レジスタ

1. 2, 1. 5 ローカルレジスタ

ステップ2. 1 ～ステップ2. 7 バリア同期処理フローチャート

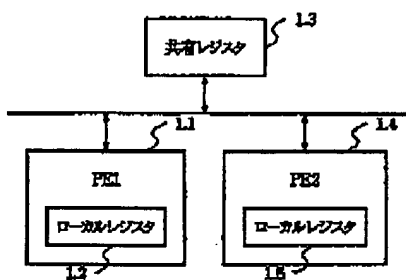
【 図4】



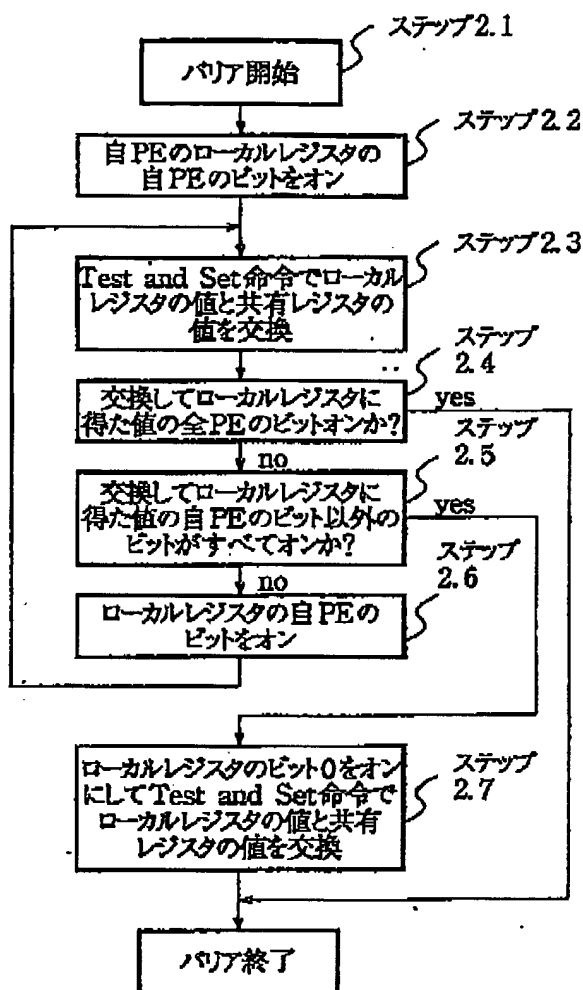
(4)

特開平9-44460

【 図1 】



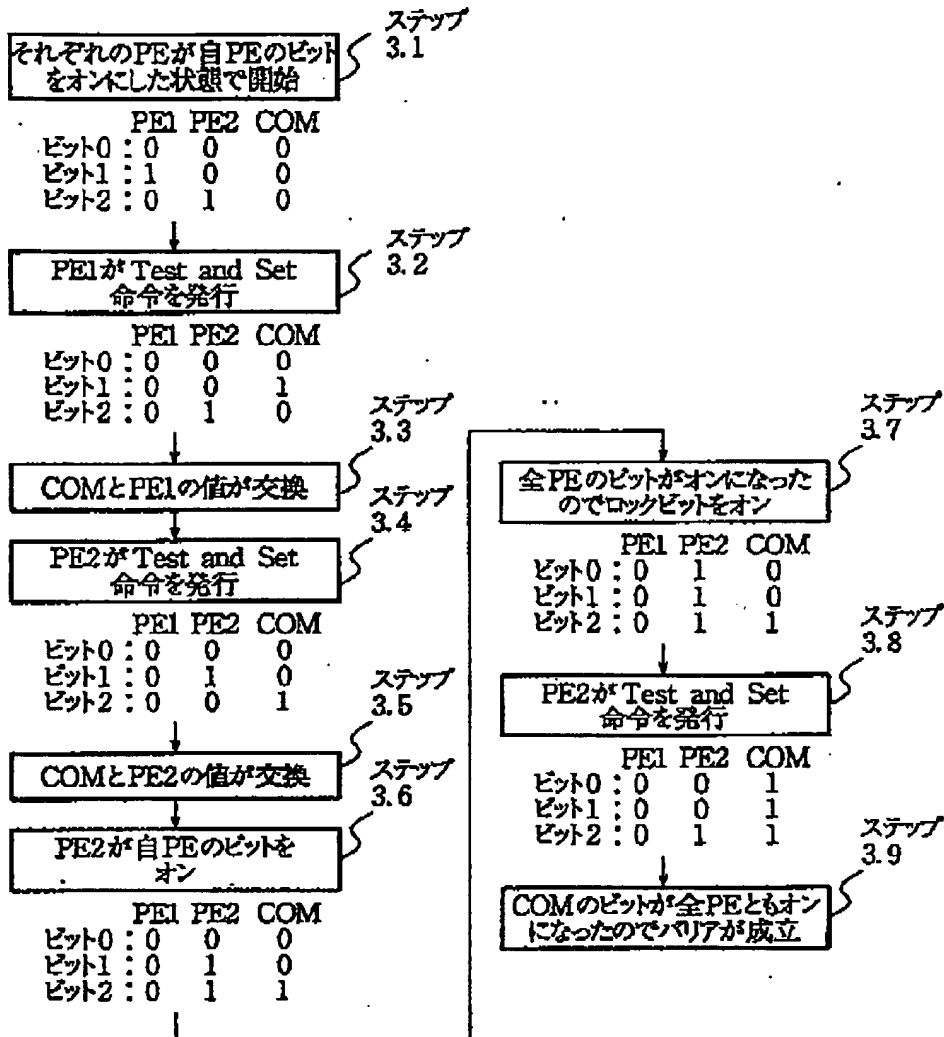
【 図2 】



(5)

特開平9-44460

【 図3 】



(6)

特開平9-44460

【 図5 】

